An International Journal
computers & mathematics with applications

PERGAMON Computers and Mathematics with Applications 46 (2003) 479–492
www.elsevier.com/locate/camwa

An Interactive Morse Code Emulation Management System

CHENG-HONG YANG
Department of Electronic Engineering
National Kaohsiung University of Applied Sciences
Kaohsiung, Taiwan 807, R.O.C.
chyang@cc.kuas.edu.tw

(Received March 2001; revised and accepted September 2001)

Abstract—Assistive technology (AT) is becoming increasingly important in improving mobility, language, and learning capabilities of persons who have disabilities enabling them to function independently and to improve their social opportunities. Morse code has been shown to be a valuable tool in assistive technology, augmentative and alternative communication, rehabilitation, and education, as well as adapted computer access methods via special software programs, hardware devices, and switches. In this study, we designed and implemented an interactive multimedia Morse code typing emulation system for persons with disabilities, which include three types of switches, a single switch, a double switch, and six switches, with three adaptive recognition methods. After practicing on the proposed system repeatedly, three people with disabilities were able to familiarize themselves with Morse code operation. Experimental results showed that participants improved in both the speed variation coefficient and ratio variation coefficient for entry of keystroke input to the computer.

Keywords—Morse code, Assistive technology (AT), Keyboard emulation, Augmentative and alternative communication (AAC).

1. INTRODUCTION

Computers, accompanied by the rapid growth in information technology, are widely used in a variety of fields and are emerging as a major trend. Most products manufactured now, however, are designed for general users, and thus, are often inaccessible to users with disabilities. Often, persons who have disabilities need auxiliary tools to use the technological devices originally designed for general users. As technology advances dramatically, these adaptive tools will play a more prominent role in their lives. Of all the adaptive devices, the ones that assist with computer input may be most important. For persons with impaired hand coordination and dexterity, the conventional computer keyboard cannot provide a typically useful communication method. Nonetheless, users who may be severely disabled require a method for entering text into a computer or word-processor for purposes of augmentative and alternative communication

This work was supported by the National Science Council (NSC) R.O.C., under contract NSC 89-2614-E-151-001. The introduced system has won the second prize in a creative competition about adaptive devices for disabled users held by the NSC in 1999.

PII: S0898-1221(03)00225-6

(AAC) and adapted computer access in their daily lives. Many computer-assisted adapted direct-selection methods and others were developed for the disabled, such as a head mouse, mini-keyboard, king-sized keyboard, trackball, joystick, alternative keyboards, keyguards, and touch screens [1].

Morse code can be transmitted as a tone-silent time series among other methods. A dot, represented as a short beep, or a dash, represented as a longer beep, are defined as tone intervals (switch down). A dot-space, which is a short pause between dots and dashes, a character space, which is a longer pause between characters, and a word-space, are defined as silent intervals (switch up). Subsequently, Morse code is simple, and can be transmitted using just a single switch. Therefore, Morse code has been shown to be an excellent candidate in AAC and assistive technology (AT), including mobility, environmental control, and adapted worksite access [2–9]. To date, over 30 manufactures/developers of Morse code input hardware or software for use in AAC and AT have been identified [10]. A variety of software packages are developed to learn Morse code, such as SuperMorse [11], NuMorse [12], Morse Code WSKE II [13], Morsek [14], Stormy Weather's Morse [15], and others [10]. None of them, however, support three different types of switches (single switch, double switch, six switch) and include management reports. Therefore, in this paper, we present an interactive Morse code emulation system to help beginners or persons who have challenges and disabilities to become acquainted with Morse code, and to communicative with a computer by implementing a multimedia package. Three test participants, two suffering from cerebral palsy (athetoid type) and one suffering from incomplete quadriparesis, achieved marked improvement in both the speed variation coefficient and ratio variation coefficient for typing. These test participants represent user groups handicapped by involuntary uncoordinated and uncontrollable movements of their limbs and/or dysfunctional movement of hands and fingers.

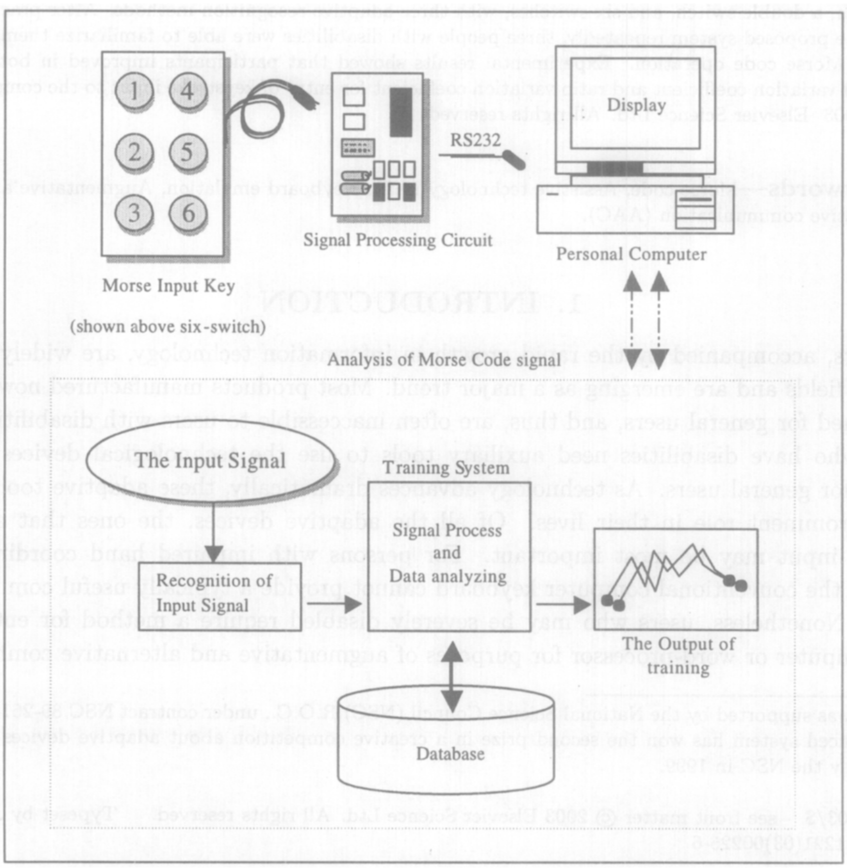


Figure 1. System frame of the Morse code emulation management system.

2. SYSTEM DESIGN

Morse code can be entered or sent with a single switch, double switch (one each for a dot or a dash), or six switches (dot, dash, dot-dot, dot-dash, dash-dot, dash-dash). A stable typing rate is strictly required in order to recognize Morse code correctly. This restriction is a major hindrance for most beginners or people with very serious disabilities to consider Morse code a useful tool. The diagram of the proposed system frame is shown in Figure 1. The signals, which are generated by a user pressing a single switch, double switch, or six switches, go through a signal processing circuit and are then sent to a personal computer (PC) via the RS232 port. In analyzing the signal, the PC first recognizes each input signal as a dot or a dash (if it is a single switch), and then translates it into a character in the recognition system. In addition to the recognition system, a training system is included, in which the individual user input speed and accuracy rate are saved in a database after every practice session. The user can view the output of the training results as well. The system design is divided into two parts: hardware and software. For the hardware part, three types of switch mode input interfaces are designed and used instead of the conventional keyboard to key in Morse code into the computer. The software part is used to receive the signals transmitted from the hardware, collecting training data, and evaluating the training performance. A detailed explanation of the hardware and software components is presented in the following sections.

2.1. Hardware Circuit Design

The hardware circuit design of the Morse code input device consists of three modules: signal processing, data communication, and key-in control (Figure 2). An 8051 single chip is adopted to handle communication between the press-button processor and the personal computer (Figure 3). Although it only has a small memory capacity and I/O, not comparable to a typical PC, it is still enough to control the device.

The 8051 internal serial communication functions are used for transmitting and receiving data. To achieve the data communication at both ends, the two pins, $T \times D$ and $R \times D$, are connected

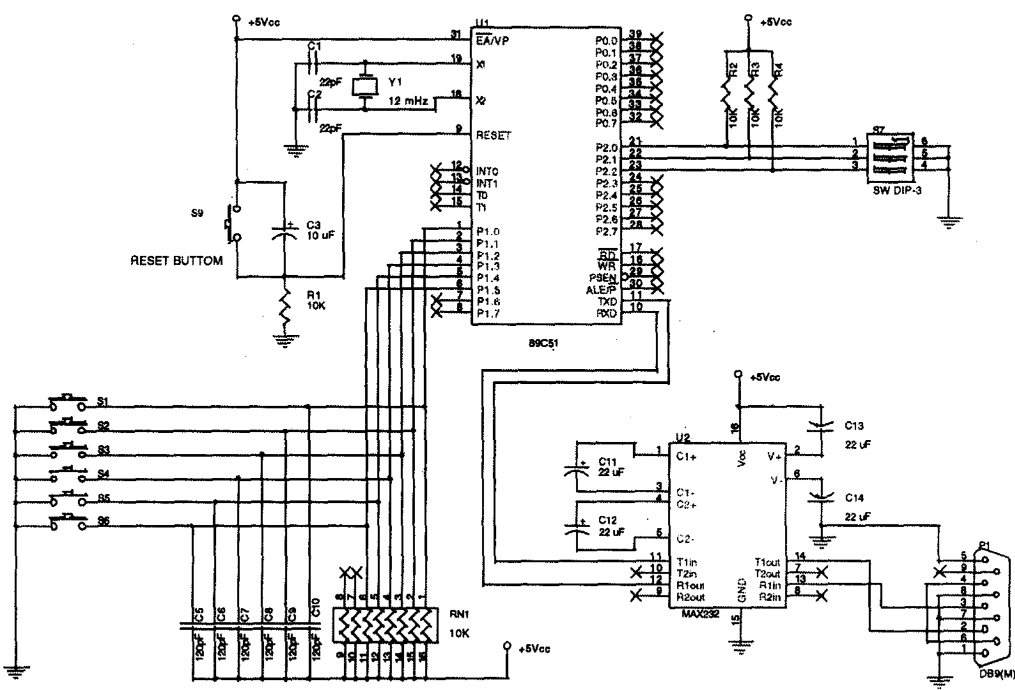


Figure 2. The circuit diagram of the Morse code input device.

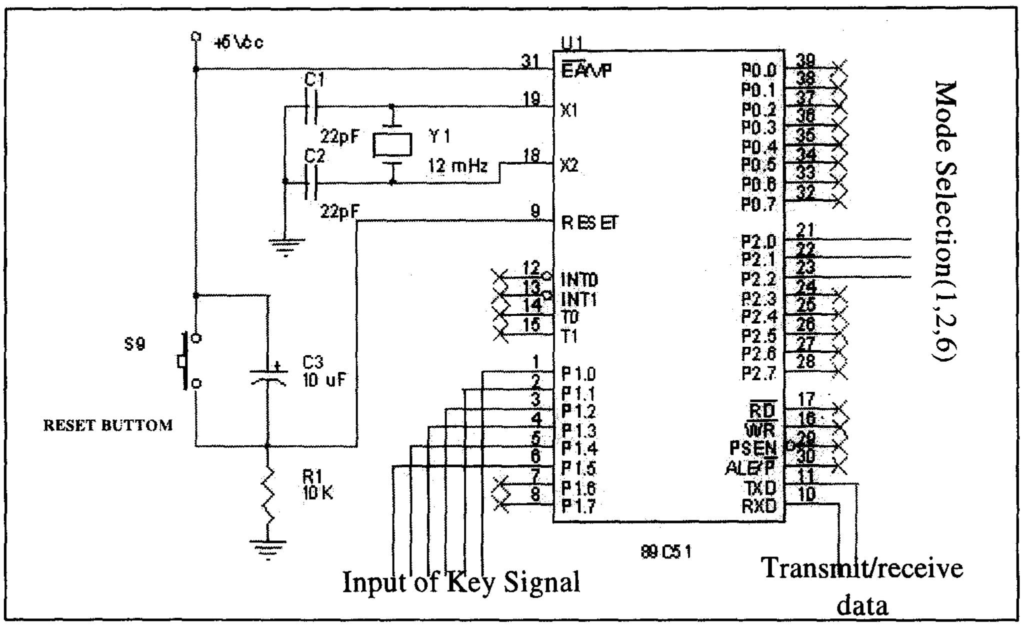


Figure 3. Key-in input signal processing diagram.

to the $T \times D$ and $R \times D$ pins of a RS-232 connector. Then, the two pins are connected to the $R \times D$ and $T \times D$ of a UART (Universal Asynchronous Receiver Transmitter) controller on the PC transmitter by a Max232 interface (Figure 4). For the data communication protocol an asynchronous transmission, with a baud rate of 9600 bps, 8 data bits, one stop bit, and nonparity check, has been adopted. The pins P2.0, P2.1, and P2.2 of 89C51 were used to select the input

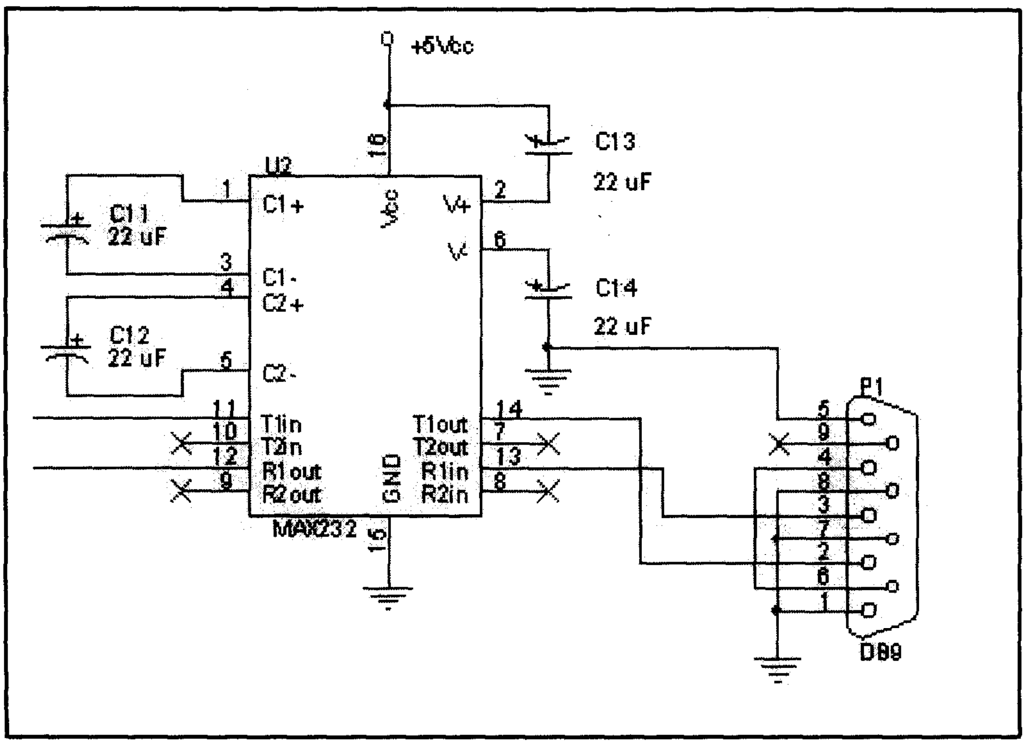


Figure 4. Communication interface circuit.

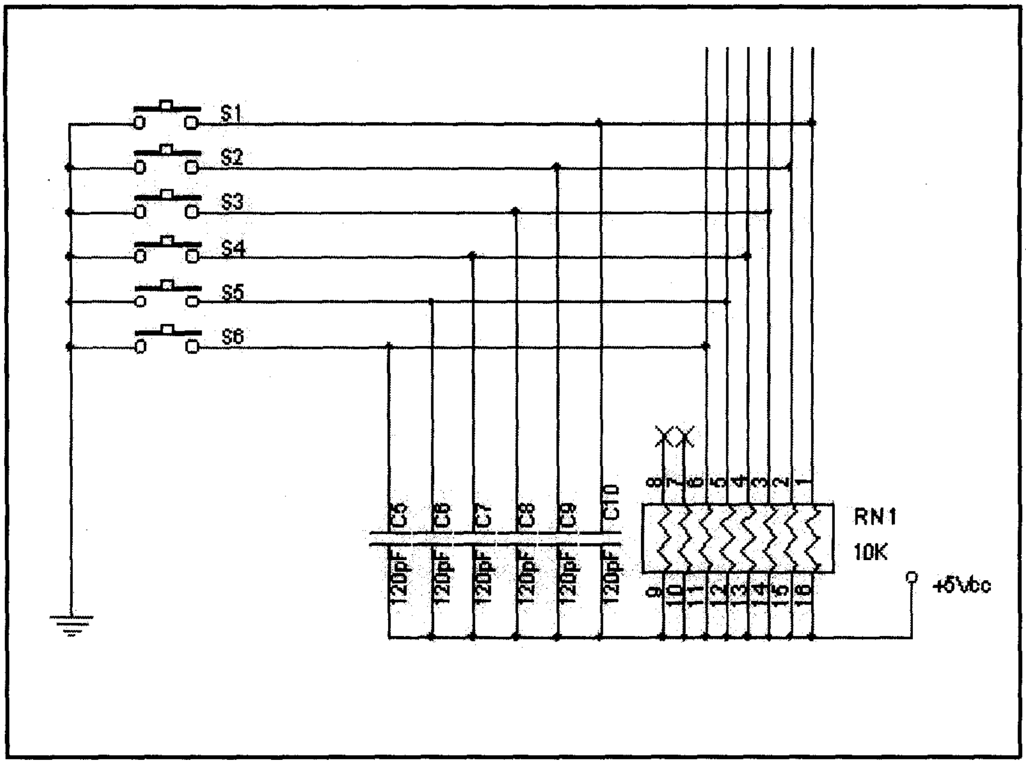


Figure 5. Key-in sensor circuit diagram.

switch mode as single switch, double switch, and six switches, respectively. Also, the pins P1.0–P1.5 were used to determine the key-in status of the user input (Figure 5).

An initialization command is added in the hardware circuit. Only after receiving the initialization command sent from the computer will the circuit start to process the key-press operations. After receiving the initialization command, the circuit will reset the device first by clearing all key-press status and data registers, and then send a signal to the PC end to indicate that the device is ready. Afterwards, all operations of the Morse code press key will be monitored. When a key is triggered (pressed or released), a program in the 8051 single chip will scan to identify the triggered key. Then, the corresponding code and the data combination of key press or release will form a byte data, which will be sent to the data register in the serial port. When data arrives at the data register in the serial port, the transmission function will be enabled. As the serial port begins to transmit data, the parallel data in the data register will be transformed into serial data. These serial data will be sent to the $R \times D$ pin of the PC server through a $T \times D$

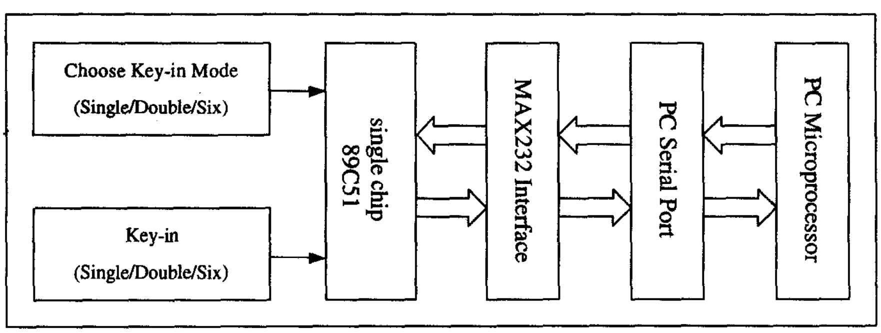


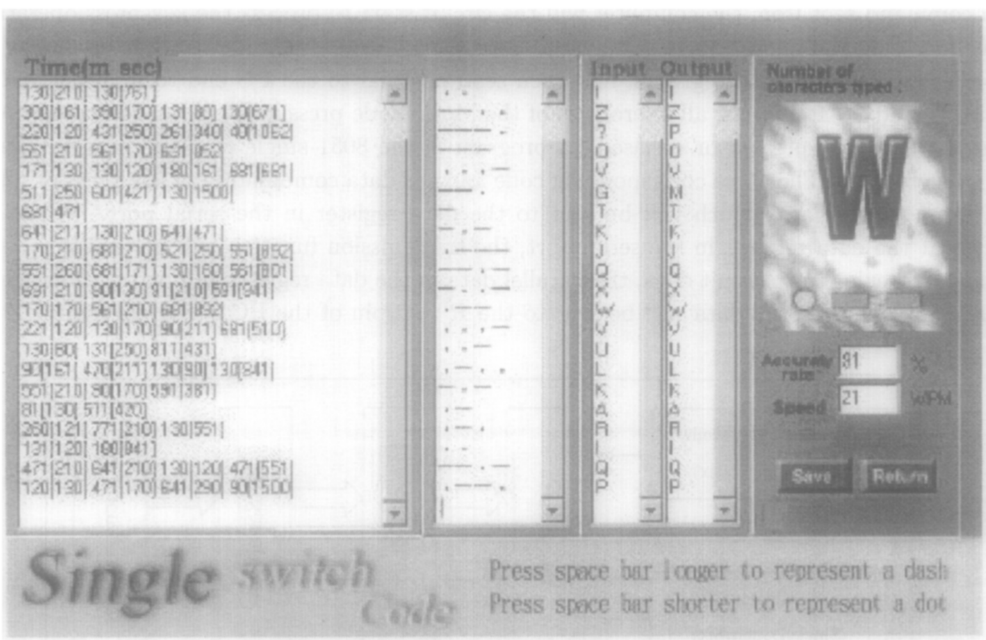
Figure 6. Signal control diagram.

signal line. After these signals are received by the UART controller of the PC, the UART will first decode them into character data, save them in a data register, and then inform the CPU that there are incoming data. After being received by the software, the data will be transmitted to the hardware end to indicate that the data has been received and the next data can be sent. Now, the circuit will begin to process the next key press or release operation (Figure 6). As for the design of the connection between the interface and the personal computer, a standard RS-232 serial port connects the MAX232 interface with the PS's microprocessor.

2.2. Software Functions

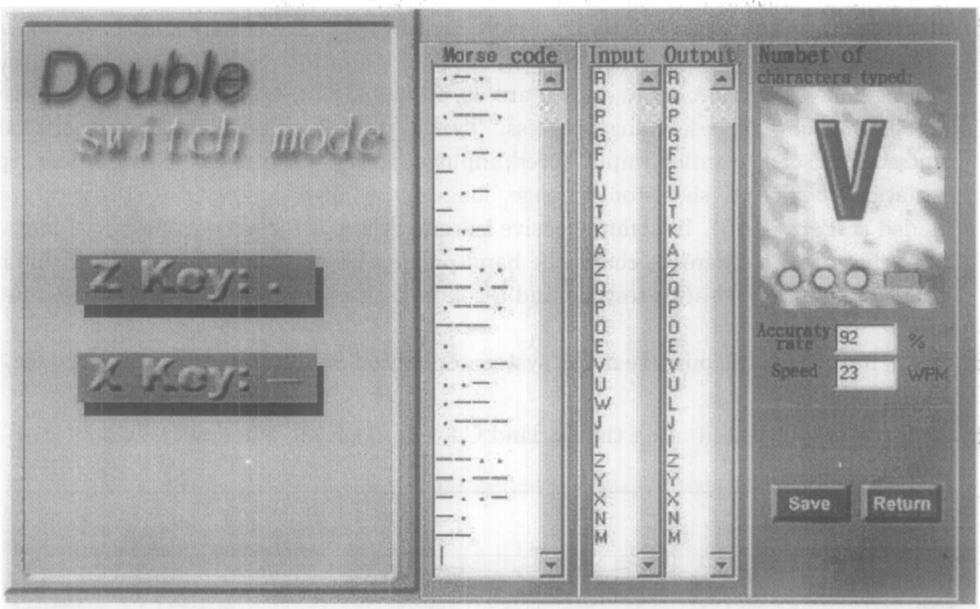
There are six major functions incorporated in this system.

(1) Files. Personalized user record files, which contain basic information of users, such as name, sex, the most often used hand before, and occupation, can be created and opened. In addition, learning results are saved in files so as to evaluate a user's learning performance.
(2) Typing selection. Three types of operations, single switch, double switch, or six switches, are provided for practice. Three types of switch operation modes are shown in Figures 7a–7c.
(3) Recognition method. Three different recognition methods, Luo and Shih (LS) [2], Shih and Luo (SL) [3], and Yang [16], are incorporated. Method 1, the LS method, proposes a system that can recognize varying speeds using an adaptive technique, the least-mean-square (LMS) algorithm. This system can adjust its characteristics to successfully recognize a message under unstable typing conditions, but the typing speed variation is limited to a range between 0.67 and two times the current speed. To satisfy this limitation, a person has to be well trained; otherwise, the system will not successfully recognize the Morse code message. However, this restriction cannot always be complied with by a beginner or by someone with severe disabilities. Therefore, this method cannot be effectively used. Subsequently, method 2, the SL method, is an improved method that combines the least-mean-square algorithm with a character-by-character matching technique to overcome

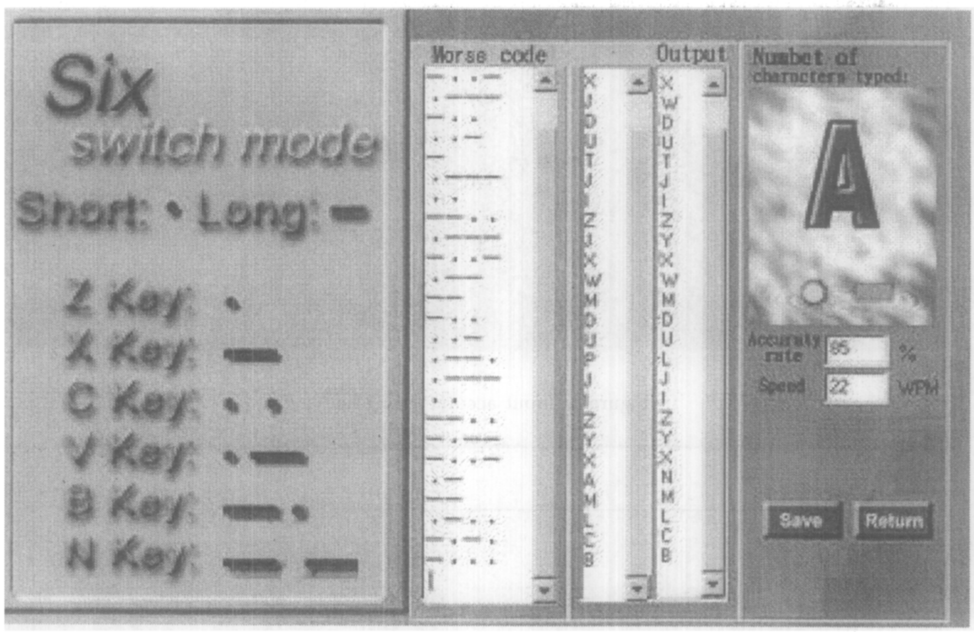


(a) Single-switch mode.

Figure 7.



(b) Double-switch mode.



(c) Six-switch mode.

Figure 7. (cont.)

the typing speed limitation. Method 3 [16] is a method in which collected predecessors of tone values and silent values are used to determine the critical tone element value or silent element value, respectively. The determined critical tone element is then used to distinguish the next incoming element as a dot or dash. The same procedure is used to discriminate the silent element.

(4) Data files. There are three different types of exercise files: preprogrammed default files, in the system, randomly computer-generated files, and customizable input files. The system will show the accuracy and speed of key press or release after each user typing operation.

(5) Statistical results. Statistical records show the variation of accuracy and speed, the exercise time, the speed variation coefficient, the ratio variation coefficient, and the variation of long tones and short tones, which are all presented in a bar or line diagrams, so that users can follow their learning progress. Two items are included in these statistical results: in personal learning, input speed, input accuracy, ratio variation coefficient, speed variation coefficient, short-tone change, long-tone change, and overall statistics are included (Figures 8–11). In comprehensive learning, there is a diagnosis chart, an education level chart, a sex difference chart, the hand-preferably-used before testing chart, and the hand-preferably-used after testing, and a comparison chart showing statistics of different users.

(6) Help. Introduces the purpose of the system, operation method, content of input data, and output results.

The system is implemented using the Borland C++ Builder 4.0, Windows version.

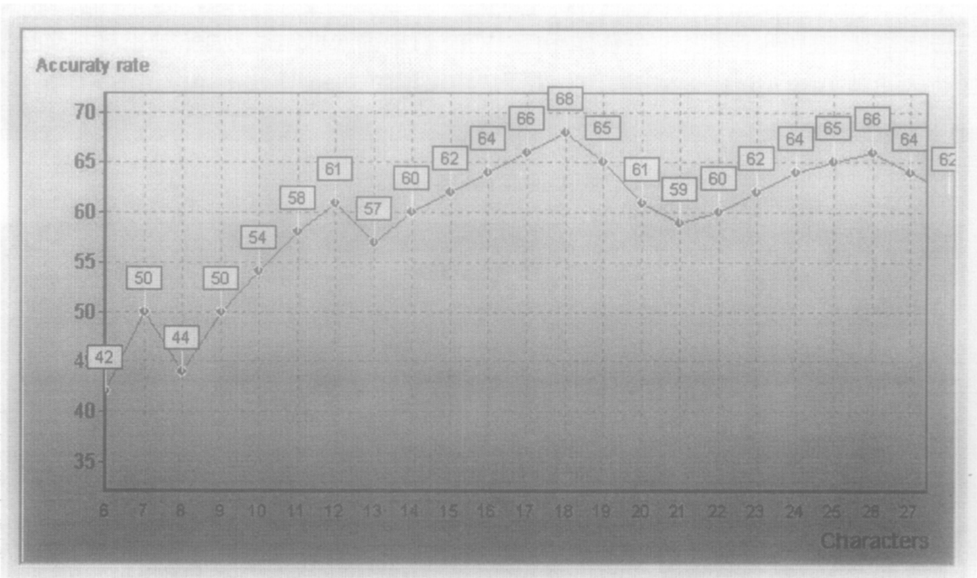


Figure 8. Input accuracy variation.

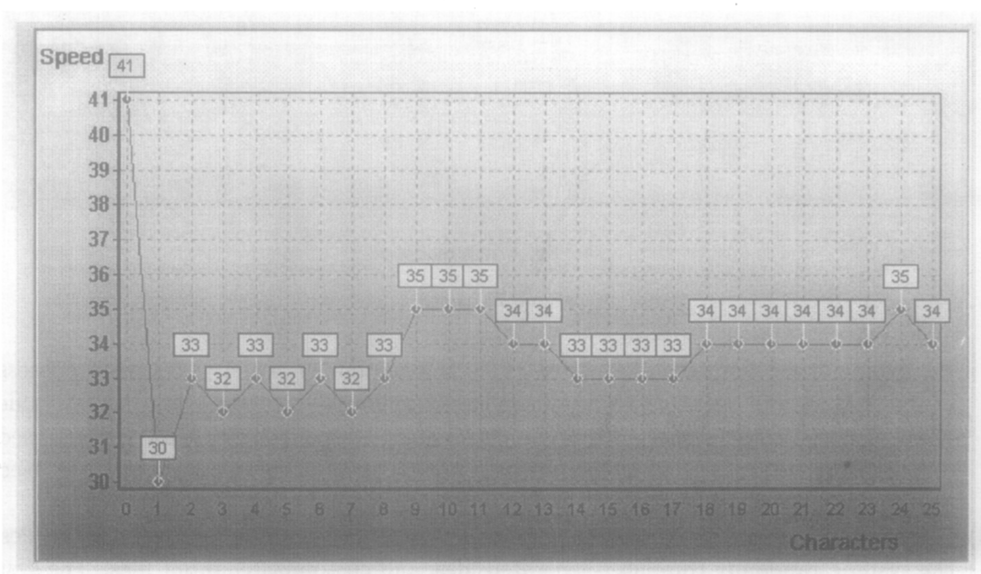


Figure 9. Input speed variation.

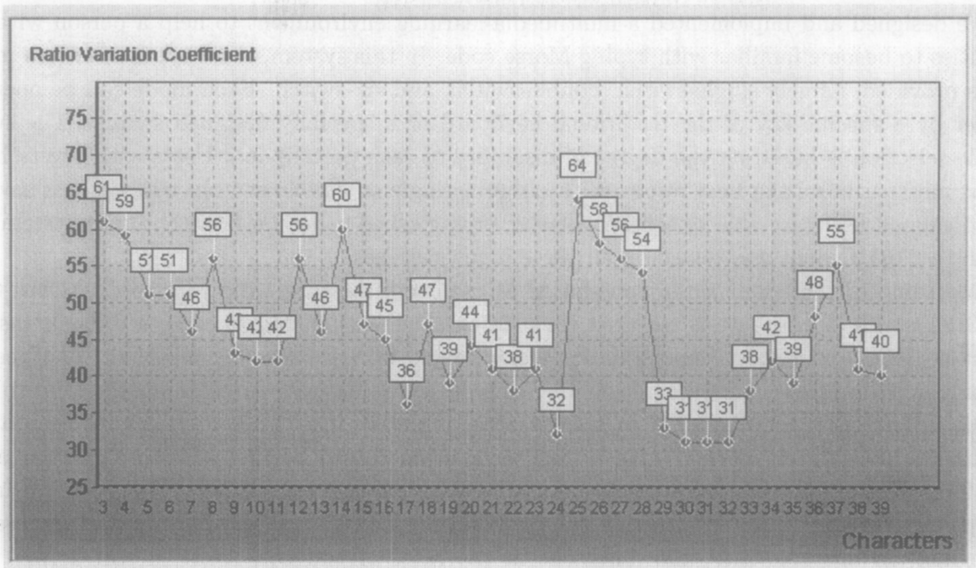


Figure 10. Ratio variation coefficient.

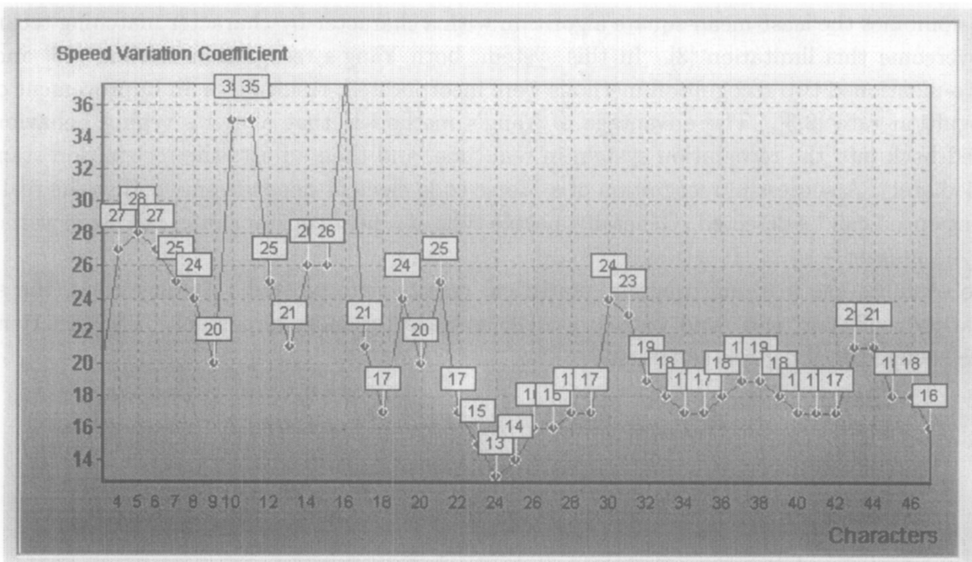


Figure 11. Speed variation coefficient.

3. RESULTS AND DISCUSSION

Many individuals with motoric and/or sensory disabilities are using newly-developed adapted-access software programs, hardware peripherals, and learning methods to help them use microprocessor devices via Morse code input systems through switches external to the computer [20]. People with limited movement or sensory capabilities have been shown to successfully operate computers and other devices via adapted switching mechanisms and Morse code emulation of keyboard input functions [10,17–20]. Research and clinical experience are indicating that the fast rate of entry and low level of physical exertion inherent in a Morse code input system make it a viable and competitive method of microprocessor control for persons who are disabled. These alternative computer access methods enable disabled people to speak, write, type, dial, create graphics, compose music, and give them the means to successfully master other modes of personal expression [10].

We designed and implemented a multimedia-learning environment to help a person with disabilities to become familiar with typing Morse code. In this system, three types of switch modes were provided, namely single-switch, double-switch, and six-switch. Each mode can be operated either by a specific key of the traditional keyboard or a specially designed switch. The switch mode selection is based on the degree of disability of each participant. Users with severe handicaps might achieve the best results with either a single or double switch, whereas less severely handicapped users are able to achieve a faster typing speed using a six-switch input system [21]. Each test participant was tested on all three input switches.

According to the definition of time-based Morse code, the tone ratio (dot to dash) has to be 1 : 3. A stable typing rate is required when using single-switch Morse code, a difficult task for most persons, and almost impossible for many people who are severely disabled. The possible error sources in Morse code recognition may include unstable typing rate, an incorrect dot to dash ratio, and an incorrect character to noncharacter interval ratio. Therefore, a suitable adaptive automatic recognition method with a high recognition rate is needed. Three high recognition methods were provided in this system that can be used to recognize varying typing speeds using an adaptive technique [2,3,16]. Although the method proposed by Luo and Shih [2] can successfully recognize an unstable rate, the difference in the rates of two characters is still limited to a range between 0.67 and 2.00 of the current typing speed. Maintaining this range would be difficult in practical applications. Subsequently, Shih and Luo proposed an improved method that combines the least-mean-square algorithm with a character-by-character matching technique to overcome this limitation [3]. In this system, both Yang's recognition method [16] and the above-mentioned two recognition methods were incorporated, resulting in an improvement of the recognition rate [2,3]. The advantage of Yang's method is that a user's typing behavior can be fed back into the recognition system in real time, and then adjusts the recognition standard immediately. A successful recognition of a Morse code element depends on whether the real time typing speed can be detected. Generally, people type at a near constant rate, but slight variations may be present.

To evaluate the learning progress, statistical results were plotted. Among them, the speed variation coefficient and ratio variation coefficient are the most important. The equations for these coefficients are given as follows:

$$V_{\text{speed}} = \frac{(\sigma_{ST}/m_{ST} + \sigma_{SS}/m_{SS})}{2} \times 100\%, \tag{1}$$

$$V_{\text{ratio}} = \frac{(\sigma_{RT}/m_{RT} + \sigma_{RS}/m_{RS})}{2} \times 100\%, \tag{2}$$

where

σ_{ST} : the standard deviation of short tones,

m_{ST} : the mean of short tones,

σ_{SS} : the standard deviation of short intervals,

m_{SS} : the mean of short intervals,

σ_{RT} : the standard deviation of long to short tone ratios,

m_{RT} : the mean of long to short tones,

σ_{RS} : the standard deviation of long to short interval ratios,

m_{RS} : the mean of long to short intervals.

From the "number-of-keystrokes" aspect, an entry interface device with more switches is preferable, since the keystroke number is reduced. From a "comprehensibility" aspect, a six-switch Morse code keyboard is the more easily memorized. From a "skill" aspect, the fewer the switches, the more stable are the tone and silent ratios. From a "fatigue-by-repetitive-activity" aspect, a keyboard with fewer switches is not ideal, since a user gets tired by maintaining stable tone and silent ratios because of the necessity of numerous keystrokes. From a "fatigue-by-other-factors" aspect, the different keyboards do not show different effects. From a "adaptability" aspect, a single-switch and six-switch Morse code input system can be better adapted than a double switch Morse code input system.

A user of this proposed method can get acquainted with the operations easily and evaluate his/her typing speed. The major difference of the system presented here compared to a system from the literature is that the user who is starting to type does not need to set a default typing speed. The system adapts to the typing speed automatically, based on the user's preceding typing speed. This increases the system's practical applications and accuracy. Furthermore, the system tells the user the characters he or she commonly mistypes. It also saves individual session

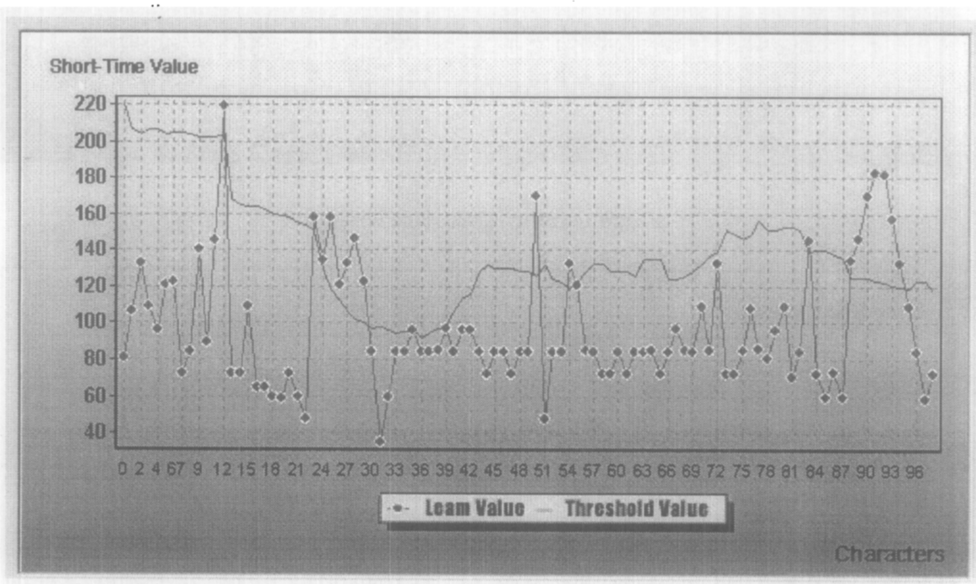


Figure 12. Short-tone value variation.

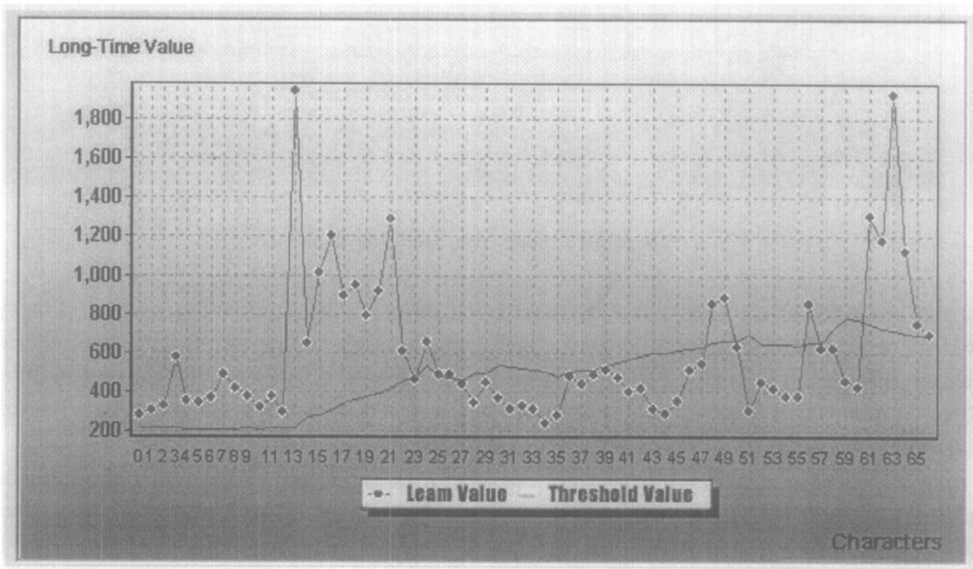


Figure 13. Long-tone value variation.

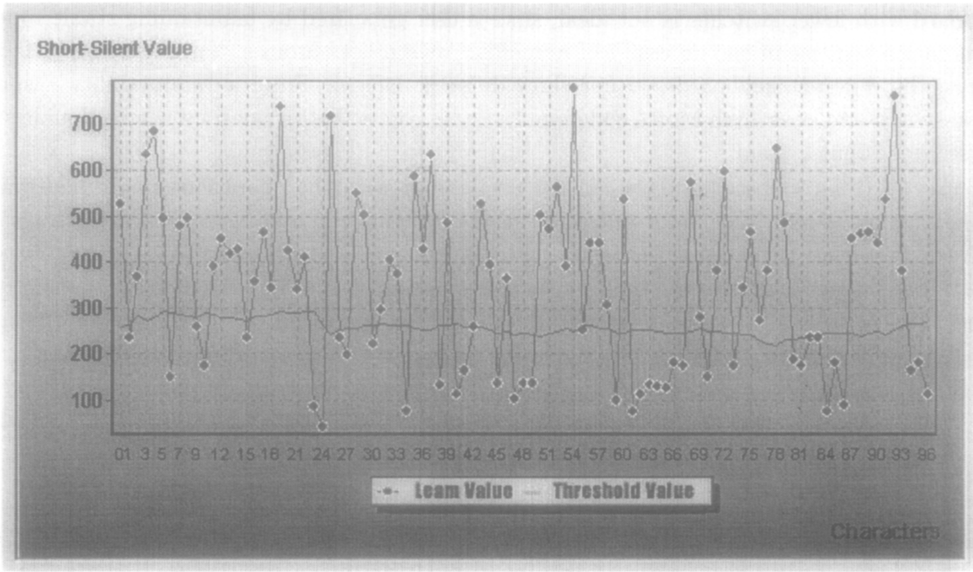


Figure 14. Short-silent value variation.

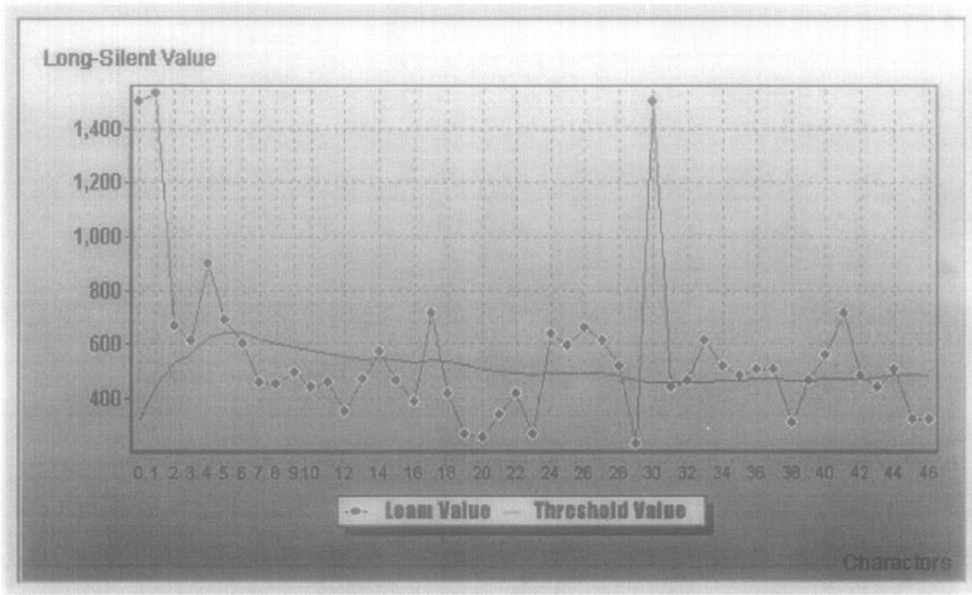


Figure 15. Long-silent value variation.

data and identifies characters that were most often mistyped in a particular session. The system provides dot and dash statistical figures, which show a user's typing speed and its prediction for each user threshold (Figures 12 to 14). Based on this graph, a user can determine the difference or error between his typing speed and the predicted speed, which will help him in adjusting the initial typing threshold.

Three people who have disabilities were tested. Participant 1 was a 14 year-old female adolescent, diagnosed with cerebral palsy, athetoid type, who exhibited involuntary and uncontrollable movement of her four limbs. Participant 2 was a 40-year-old male who suffered a spinal cord injury with incomplete quadriparesis. His right wrist cannot be extended purposefully and individual finger movement is limited, resulting in severely dysfunctional hand movement. Participant 3 was a 14-year-old male who has been diagnosed with cerebral palsy. His voluntary movements were accessible but an initial delay was exhibited before the movement was initiated. Each person typed 50 characters in a test for a total of six runs during a two-month period. After repetitive training sessions the typing speed of the above users progressively improved for all of the three different switches input systems. Yang's method was used to evaluate the test participant's progress. Typing speed improved by about 11% for a single switch, 13% for double switch, and 16% for a six-switch input system. According to the training results, the ratio variation coefficient was larger than the speed variation coefficient. That means that it was more difficult to maintain a stable ratio than a stable speed. After two months' practice, the speed variation coefficient and ratio variation coefficients had improved from 54 down to 32 and from 34 down to 26 for single-switch entry of Morse code to the system.

4. CONCLUSION

In this paper, an easy-to-operate learning system for Morse code was specifically designed for people with such physical disabilities as muscle atrophy, athetoid cerebral palsy, and other severe handicaps. In addition to the implementation of a user-friendly interface, three Morse code recognition rate methods were incorporated in this system to discriminate Morse code typing. A reliable Morse code recognition method can be important for some disabled persons, helping to compensate for an unstable input rate, and correcting user errors resulting from long periods at the computer. Experimental suggestions prove that the system can improve the speed variation coefficient and ratio variation coefficient. We submit that persons with disabilities will be able to use many computer applications more quickly with assistance from this system, thus increasing their communication abilities. Also, Morse code input to activate mobility and environmental control devices may open worlds of educational, vocational, and recreational opportunities to many persons previously shut out of these pursuits [10].

REFERENCES

1. R. Bower *et al.*, Editors, *The Trace Resourcebook—Assistive Technology for Communication, Control, and Computer Access*, Trace Research and Development Center, Universities of Wisconsin-Madison, Waisman Center, (1998).
2. C.H. Luo and C.H. Shih, Adaptive Morse-coded single-switch communication system for the disabled, *Int. J. of Biomed. Comput.* **41**, 99–106, (1996).
3. C.H. Shih and C.H. Luo, A Morse-coded recognition system with LMS and matching algorithms for persons with disabilities, *Int. J. of Medical Informatics* **44**, 193–202, (1997).
4. S.P. Levine, J.P. Gauger, L.D. Bowers and K.J. Khan, A comparison of mouth stick and Morse code text inputs, *Augmentative and Alternative Communication* **2**, 51–55, (1986).
5. L.N. Goble and H.A. Colle, High-speed Morse code training, *Proceedings of the IEEE 1985 National Aerospace and Electronics Conference, NAECON*, 944–951, (1985).
6. D.W. Lywooed and J.J. Vasa, Computer-terminal operating and communication aid for the severely handicapped, *Medical and Biological Engineering* **12**, 693–695, (1974).
7. D.A. Shannon, W.S. Staewen, J.T. Miller and B.S. Cohen, Morse code controlled computer aid for the nonvocal quadriplegic, *Medical Instrumentation* **15** (5), 341–343, (1981).
8. A. Thomas, Communication devices for the non-vocal disabled, *Computer* **14**, 25–30, (1981).
9. J.J. French, F. Silverstein and A.A. Siebens, An inexpensive computer based Morse code system, In *Proceedings of the RESNA 9^{th} Annual Conference*, Minneapolis, MN, pp. 259–261, (1981).
10. T.W. King, *Modern Morse Code in Rehabilitation and Education*, Allyn and Bacon, Boston, MA, (1999).
11. D.G. Finley, *Morse Code: Breaking the Barrier*, MFJ Enterprises, (1995).
12. T. Lacy, *NuMorse*, Wolverhampton, U.K., (1994).
13. Words+, *Morse Code WSKE II*, Palmdale, CA, (1995).
14. Kinetic Designs, *Morsek*, Olalla, WA, (1994).
15. *Stormy Weather's Morse Code Ltd*, Ontario, Canada, (1996).
16. C.-H. Yang, A novel approach to adaptive Morse code recognition for disabled persons, *Mathematics and Computers in Simulation* **54**, 23–32, (2000).
17. J. Lynds, Learning Morse, *Morsels* **2** (2), 2, (1996).
18. F. Romney and D. Romney, *Why Morse—And How? Shareware Icons and Documentation*, Clarksville, IN, (1987).
19. J. Eulenberg, Augmentative and alternative communication and assistive technology expert, Communication via e-mail on Morse 2000 list-server, (1997).

20. T.W. King, *Assistive Technology: Essential Human Factors*, Allyn and Bacon, Boston, MA, (1999).
21. C.-H. Luo and M.-C. Hsieh, Morse code typing training of a teenager with cerebral palsy using microcomputer technology, *Augmentative and Alternative Communication* **15**, 1–6, (1999).